CONSIDERATIONS REGARDING INFORMATIONAL MANAGEMENT OF PRODUCT STRUCTURE

Alexandru SZEL[1], Macedon GANEA[2], Radu Ţarcă[3] , Tiberiu VESSELENY[4], Florin BLAGA[5]

[1]University of Oradea,szel.alexandru@yahoo.com

[2]University of Oradea, macedonganea@yahoo.com

[3]University of Oradea, rctarca@yahoo.com

[4]University of Oradea, vtiberiu@uoradea.ro

[5]University of Oradea, fblaga@uoradea.ro

***Abstract*—** The paper presents a model for product structure management, highlighting the structure of data collections and also their use. The data structure query procedure is also presented, which uses a special program competitive at international level. The solution supports the computer assisted manufacturing in the preparation stage as well as in the operation stage of the processing, automatically computing the quantity of components needed to be manufactured for the given product quantities.

***Keywords*—** access, command, level, structure.

I. INTRODUCTION

WE aimed to define an easy to use minimal structure, which also ensures a minimal operation time and allows an on-line use with very rapid response times.

The adopted solutions have a high degree of generality and can be used in any discrete type production systems (machine tools, textile industry food industry, etc.) [1].

II. DATA STRUCTURE DEFINITION

For product structure and components [2]-[4] we define a relational arborescent type database which describes the product structures by level, as for example:

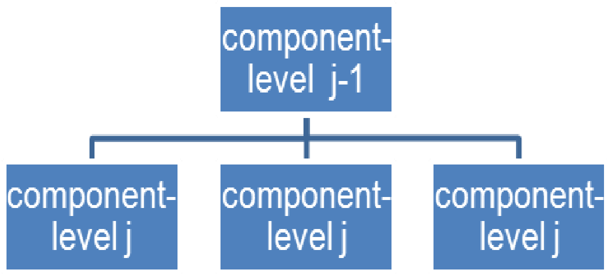

Fig. 1. Product structure tree

Storage takes place in a structure file named STR:

STR.DBF

SCOMPOUND N 13

SCOMPONENT N 13

SPIECESLEV N 10.3

SDECOMP C 1 (Y, N)

This contains the codes of the compound, components, quantity of components in the compound and also a logic indicator Y/N, which shows whether the component at it's turn can be decomposed in other components (end of chain indicator).

These data structures can be represented as relational tables.

An example for such a relational database with product structures is presented below:

TABLE I
EXAMPLE OF STRUCTURES WITH QUANTITIES ON LEVEL

COMPOUND CODE 1								LEV.0
2		3		4		5		LEV.1
6	7	8	9	10	11	12	13	LEV.2

Given the complexity of the products in practice (sometimes even with more than ten thousand components), for the computer assisted fabrication it is necessary the development of a high-speed program which allows us to query the constructive structure.

The interrogation mode of the structure can be random using the key SCOMPOUND, and sequential as SCOMPOUND + SCOMPONENT.

TABLE II
RELATIONAL TABLE OF THE DATABASE

SCOMPOUND	SCOMPONENT	SPIECESLEV	SDECOMP
1	2	2	Y
1	3	3	Y
1	4	2	Y
1	5	4	Y
2	6	2	N
2	7	3	N
3	8	2	N
3	9	3	N
4	10	2	N
4	11	3	N
5	12	2	N
5	13	3	N

III. Interrogation Of Tree Database

From the interrogation point of view we start from a file named “ORDER.DBF” which contains the demanded products, with the product code and quantities to be produced[1], [5]-[6]:

ORDER.DBF
CPRODCODE N 13
CQUANT N 9.3

The results of the interrogation will be stored in a file named “OUTPUT.DBF” containing the component code and other information (level of structure, quantity on level, end of chain indicator, linked component code).

OUTPUT.DBF
ICOMPONCODE N 13
IQUANTI 9.3 N
ILEVEL N4
IQUANTLEV 10.3 N
IDECOMP C 1 (Y, N)
ICOMPOUND N 13

The annex 1 contains the query program in operational state. The algorithm is working with minimal read/write from or to external files, by its simplicity, rivaling with the most advanced solutions.

Reading sequentially the “ORDER” file for each CPRODCODE code, we began random access in the STR file, which then will be read sequentially until SCOMPOUND equals CPRODCODE with all the branches of the tree which satisfies the above conditions and which has as descendants (SDECOMP=”D’’ We load a stack memory and compute the quantities which are to be produced until SDECOMP≠”N”).

After that we read from the stack and write to the output file OUTPUT all the components until each article from the stack will be processed. The algorithm continues with another product code from the “ORDER” file until the end of this file.

Some practical tests had proven that a stack depth of 400 is sufficient.

The high speed of the algorithm allows its use even in the pre-planning phase, at computing the necessary quantity to be produced, at computing the manpower and the raw materials.

The program from annex 1 ensures the decomposition of the product structure. This high-speed algorithm is internationally competitive.

IV. Annex No 1, Tree Database Interrogation Program

```
CLOSE ALL
CLEAR
SET TALK OFF
WW=' '
DO WHILE WW#’Y’.AND.WW#'N'
    @4,4 SAY ' Query the database (Y/N):' GET WW
    PICT 'X'
READ
ENDDO
IF WW=‘Y’
MCOMPOUND=0
MQUANT=0
MLEVEL=0
PCOMPOUND=0
P=1
F=0
I=1
DECLARE SPT(400)
DECLARE VCOMPON(400)
DECLARE VQUANT(400)
DECLARE VDECOMP(400)
DECLARE VLEVEL(400)
DECLARE VSTERIL(400)
DECLARE VCOMPOUND(400)
DECLARE VQUANTLEV(400)
DO WHILE I<401
MCOMPON(I)=0
SPT(I)=0
VDECOMP(I)='N'
VLEVEL(I)=0
VSTERIL(I)=' '
VCOMPOUND(I)=0
VQUANTLEV(I)=0
I=I+1
ENDDO
USE OUTPUT EXCLUSIVE
ZAP
USE ORDER CPRODCODE IN 1
USE STR ORDER STR IN 2
USE OUTPUT IN 3
SELE 1
DO WHILE .NOT.EOF()
WPT=0
NRCP=0
NRPT=0
 MCOMPOUND=CPRODCODE
 MQUANT=CQUANT
 MLEVEL=0
 P=1
```

```
 VLEVEL(P)=MLEVEL
 VDECOMP(P)=‘Y’
 VSTERIL(P)=‘Y’
 PCOMPOUND=1
 F=0
 DO WHILE P>0
 SELE 2
 SET EXACT OFF
SEEK STR(MCOMPOUND,13)
 IF .NOT.FOUND().and. VLEVEL(p)#0
 @ 4,4 SAY 'LACK OF COMPOUND IN STRUCTURE : '+STR(MCOMPOUND,13)+ STR(WPT,13)
 WAIT ' '
 VDECOMP(P)='N'
 P=0
 ELSE
 DO WHILE STR(SCOMPOUND,13)=STR(MCOMPOUND,13).AND..NOT.EOF()
P=P+1
SPT(P)=NRPT
VCOMPON(P)=SCOMPON
 VQUANT(P)=SQUANTLEV * MQUANT
VQUANTLEV(P)=SQUANTLEV
VLEVEL(P)=MLEVEL+1
 VDECOMP(P)=SDECOMP
 VCOMPOUND(P)= A->CPRODCODE
 VSTERIL(P)='N'
  SKIP
  ENDDO
  VSTERIL(PCOMPOUND)=‘Y’
  F=0
  DO WHILE P>0.AND.F=0
 IF VDECOMP(P)=‘Y’.AND.VSTERIL(P)='N'
 WPT=SPT(P)
 MCOMPOUND=VCOMPON(P)
  MQUANT=VQUANT(P)
  MLEVEL=VLEVEL(P)
  PCOMPOUND=P
  F=1
  ELSE
  SELE 3
 APPEND BLANK
REPLACE ICOMPONCODE WITH VCOMPON(P)
REPLACE IQUANTI WITH VQUANT(P)
REPLACE ILEVEL WITH VLEVEL(P)
REPLACE IDECOMP WITH VDECOMP(P)
REPLACE IQUANTLEV WITH VQUANTLEV(P)
REPLACE ICOMPOUND WITH VCOMPOUND(P)
P=P-1
ENDIF
ENDDO
ENDIF
ENDDO
SELE 1
SKIP
ENDDO
 ENDIF
 CLOSE ALL
 CLEAR
 RETURN
```
[1], [5]-[6]

CASE STUDY “ORDER” FILE [7]-[8]

CPRODCODE	CQUANT
1	10,000
2	5,000
3	3,000

STRUCTUR file

SCOMPOUND	SCOMPONENT	SQUANTLEV	SDECOMP
1	2	2,000	Y
1	3	3,000	Y
1	4	2,000	Y
1	5	4,000	Y
2	6	2,000	N
2	7	3,000	N
3	8	2,000	N
3	9	3,000	N
4	10	2,000	N
4	11	3,000	N
5	12	2,000	N
5	13	3,000	Y
13	14	2,000	N
13	15	3,000	N

TABLE II
RESULTS THE OUTPUT FILE AFTER RUNNING THE PROGRAM WITH THE ABOVE MENTIONED DATA

ICOMPON	IQUANT	ILEVEL	IQUANTINIV	IDECOMP	ICOMPOUND
15	360,000	3	3,000	N	1
14	240,000	3	2,000	N	1
13	120,000	2	3,000	Y	1
12	80,000	2	2,000	N	1
5	40,000	1	4,000	Y	1
11	60,000	2	3,000	N	1
10	40,000	2	2,000	N	1
4	20,000	1	2,000	Y	1
9	90,000	2	3,000	N	1
8	60,000	2	2,000	N	1
3	30,000	1	3,000	Y	1
7	60,000	2	3,000	N	1
6	40,000	2	2,000	N	1
2	20,000	1	2,000	Y	1
7	15,000	1	3,000	N	2
6	10,000	1	2,000	N	2
9	9,000	1	3,000	N	3
8	6,000	1	2,000	N	3

CONCLUSIONS

The solution can be applied to any discrete technology (machine tools, textile industry, food industry, electric devices, and so on). Outstanding advantages can be obtained at products with many components and at series production with mixed fabrication. The simplicity of the file structure, the high speed decomposition of the product structure with the above program allow us to conduct even on-line calculations, underlying the operative management of the production.

In the first phase the program can be used to calculate the necessary production components by the tree decomposition starting from the contracted products. Followed by the calculation of the net necessary production quantity, taking into account the unfinished production stock. Estimations can be made already at this level about the end date of the production by calculating the necessary production time for the remaining quantities. Depending on the machinery calendars, the capacity, some bottlenecks (i.e. machinery numbers, number of available tools, the possibility of splitting the batch on multiple machines) the production quantity is divided in batches, as close as possible to the economic batch, which ensures the best possible loading of the machines. The launch of the manufacturing is realized on an order and batch level.

REFERENCES

[1] A. Szel, and M, Dzitac, 'Computer upgrades of Manufacturing Engineering (A gépgyártási technológiák számitógépes fejlesztése)' - *Müszaki szemle* 61/2013, EMT Kolozsvar, pp. 38-44.

[2] *fr.wikipedia.org/wiki*/Base_de_donnes_hierarchique.

[3] Varga , *Matrix Interactive Analysis*, 2000.

[4] Hardouin-Duparc, *Maxima limites d*une caractiristique de certaines transformations d*arborescences*, 1987.

[5] Naimi, *Une structure arborescente pour une classe d*algoritmes d*exclusion mutuelle*, 1987.

[6] Trhel, Proprits, *Du nombre harmonique H n lies des transformation d*arborescences*, 1986.

[7] Berge, *Theorie des graphes et ses aplications*, 1985.

[8] Fr. Kovacs, G. Cojocaru, *Roboţii în acţiune. Sisteme flexibile de fabricaţie şi fabricaţia de serie*, Ed. Facla, Timişoara, 1985.